Continuous-Time Nonlinear Systems Identification with Output Error Method Based on Derivative-Free Optimisation

M. Brunot [*,**] **A. Janot** [*] **F. Carrillo** [**]

[*] *ONERA, 2 Avenue Edouard Belin, 31055 Toulouse, France (e-mail: Mathieu.Brunot@onera.fr and Alexandre.Janot@onera.fr).*
[**] *LGP ENI Tarbes, 47 avenue d'Azereix, BP 1629, 65016 Tarbes, France(e-mail: Francisco.Carrillo@enit.fr)*

Abstract: The purpose of this study is to tackle the nonlinear system identification benchmarks proposed by (Schoukens and Noël, 2016). Two of the three benchmarks are considered, namely the cascaded tanks setup and the Bouc-Wen hysteretic system. Our approach is an output error method based on continuous time models. Due to the nonlinearities, the derivatives of the output with respect to the parameters are not defined everywhere. We compare the performance of two derivative-free optimisation solvers: the Nelder-Mead simplex and the NOMAD algorithm. Both are available in the OPTI Toolbox. The results suggest that the method is appropriate for those systems. However, it is not possible to discriminate between both optimisation solvers.

Keywords: Identification and modelling; Continuous time system estimation; Nonlinear system identification; Output error method

1. INTRODUCTION

In robotics and mechanical engineering the dynamic models are based on differential equations which often result from Newton's law or Lagrange equations. Recently, the identification of continuous-time models has grown in popularity in the field of Automatic Control (Garnier and Wang, 2008) and see the recent special issue in the International Journal of Control (Garnier and Young, 2014). The output error method is an option to deal with such problems. It consists in minimizing the difference between the simulated model output and the measured output. This approach has proven its suitability in Automatic Control (Carrillo et al., 2009), in robotics (Gautier et al., 2013) and in aeronautics (Klein and Morelli, 2006) for instance.

The aim of this paper is to evaluate if the continuous-time output error method is suitable for identifying two of the non-linear systems proposed by (Schoukens and Noël, 2016) as benchmarks for the community. We will deal with the parametric identification of the Bouc-Wen hysteretic system and the cascaded tanks setup. As it will be seen, the models are continuous but not differentiable everywhere. Thus, optimisation algorithms based on the differentiability of the cost functions cannot be employed. Two derivative-free algorithms are used and compared: the well-known Nelder-Mead simplex and the recent NOMAD optimizer, which are both available in the free OPTI Toolbox for MATLAB©.

This paper is organised as follows. Section 2 deals with the general methodology for output error identification in continuous time framework and presents the optimisation algorithms considered. In section 3, the model of the cascaded tanks setup is developed and the results are detailed. Section 4 follows the same structure for the Bouc-Wen hysteretic system. Finally, section 5 provides concluding remarks.

2. GENERAL METHODOLOGY

2.1 Continuous Time Output Error Method

With the Output Error Method (OEM), the unknown system parameters are tuned so that the simulated model output fits the measured system output. To evaluate the difference between the two outputs many criteria may be used, as explained in (Walter and Pronzato, 1997). The criterion minimisation is usually solved thanks to non-linear optimisation algorithms based on a first- or second-order Taylor series expansion. That requires the computation of the criterion derivatives with respect to the parameters. In some cases those derivatives can be exactly known, but in most cases the derivatives are approximated by finite differences.

To simulate the continuous-time system and obtain a simulated output, the differential equations must be solved. Many numerical solvers exist in the literature like the well-known Runge-Kutta method, for further examples see (Hairer et al., 1993). In this article, they will be referred as "integration solvers" to avoid confusion with the "optimisation solvers" introduced in the previous paragraph. In practice, the integration solver needs the same input as the real system and a set of values for the parameters to identify. The choice of the integration solver is decisive. For each model, the practitioner must find the integration solver which suits to the system properties. For instance, if the system presents two dynamics whose the characteristic

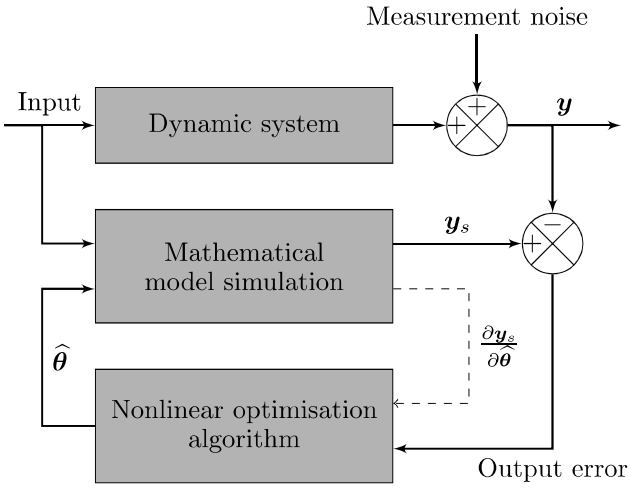

Fig. 1. Output Error Method schematic diagram

times greatly differ, a stiff solver should be employed. If the integration solver is not appropriate, it may lead to a biased identification.

The initial values is a crucial point for OEM. With a bad initialisation the optimisation solver may lead to local minimum (if it is a local optimizer) or even diverge. The integration solver may also diverge if the parameters are not suitable. Depending on the application, different techniques may be used to initialize correctly the method. If the problem is linear with respect to the parameters and if all the states are available, a Least-Squares (LS) estimation can be employed. As shown in (Gautier et al., 2013), in the field of robotics the Computer-Aided Design (CAD) values of the inertia are enough accurate to initialize. In aircraft identification, initial values can be available from wind tunnel test or computational fluid dynamics.

Inspired from (Jategaonkar, 2006), Figure 1 illustrates the OEM principle where $\boldsymbol{y}$ is the $(N_s \times 1)$ vector of the measured output, $\boldsymbol{y}_s$ is the $(N_s \times 1)$ vector of the simulated output, $\widehat{\boldsymbol{\theta}}$ the $(N_\theta \times 1)$ vector of estimated parameters, and $\frac{\partial \boldsymbol{y}_s}{\partial \widehat{\boldsymbol{\theta}}}$ is the output sensitivity, which is a $(N_s \times N_\theta)$ jacobian matrix. N_s is the number of sampling points considered and the N_θ is the number of unknown parameters. As it can be seen, the only stochastic signal is the measurement noise. The input signal is indeed assumed to be noise free. In addition, the integration solver is deterministic. It is consequently impossible to take into account process noise in the simulation. This is why the third proposed benchmark is not considered in this article.

2.2 Optimisation Solvers

OPTI Toolbox This benchmarks challenge was the opportunity to test different optimisation solvers. Our choice was to employ the OPTimization Interface (OPTI) Toolbox developed by (Currie and Wilson, 2012). This toolbox is a free interface between MATLAB© and many open source and academic solvers. From this toolbox, two derivative free algorithms have been selected to solve unconstrained nonlinear least-squares with a quadratic criterion. One noteworthy point is that, after convergence, the toolbox computes the jacobian matrix with finite differences for the statistical analysis.

Fig. 2. Cascaded tanks setup

Nelder-Mead Simplex The first considered algorithm is the well-known Nelder-Mead (NM) simplex from (Nelder and Mead, 1965). This heuristic method is based on a polytope of $N_{theta} + 1$ vertices. At each iteration, the vertice, where the cost function is the largest, is modified according to specific rules. It exists many variants of this algorithm depending on the updating rules. We decided to use the algorithm available in the open-source library NLopt developed by (Johnson, 2016) and provided in OPTI.

NOMAD Optimizer The second optimizer is the algorithm called Nonlinear Optimization by Mesh Adaptive Direct (NOMAD) search, from (Abramson et al., 2016). NOMAD is a direct search method, i.e. derivative free. At each iteration, a mesh is designed around the current optimum and the function is evaluated at any mesh points. The mesh is based on a given pattern. Thus, the choice of directions is fixed and finite. If the *search* step does not manage to find a new optimum, it is followed by the *poll* step. This second step consists of a local exploration around the optimum. For this step, the set of points to be evaluated is defined by orthogonal directions which are dense in the unit sphere. At each *poll* step, a new set is constructed. The *search* step is a common element to all Generalized Pattern Search (GPS) algorithms. The specificity of NOMAD lies on the *poll* step which gives more flexibility in the directions.

3. CASCADED TANKS

3.1 Model Description

According to (Schoukens and Noël, 2016), the model of the plant (Fig. 2) comes from Bernoulli's principle and is

given by:

$$\begin{aligned}\dot{x}_1(t) &= -k_1\sqrt{x_1(t)} + k_4 u(t) + \omega_1(t)\\ \dot{x}_2(t) &= k_2\sqrt{x_1(t)} - k_3\sqrt{x_2(t)} + \omega_2(t)\\ y(t) &= x_2(t) + e(t),\end{aligned}$$

where u is the input signal, y is the measured output signal, x_1 and x_2 are the system states, ω_1 and ω_2 are the process noises, e is the measurement noise, and k_1, k_2, k_3 and k_4 are the system parameters. As it has been said, the OEM technique relies on a deterministic numerical integration of the dynamic model. Hence, the process noises cannot be taken into account. With respect to (Wigren, 2006) the model can be written:

$$\begin{aligned}\dot{h}_1(t) &= -\frac{a_1\sqrt{2g}}{A_1}\sqrt{h_1(t)} + \frac{k_u}{A_1}u(t)\\ \dot{h}_2(t) &= \frac{a_1\sqrt{2g}}{A_2}\sqrt{h_1(t)} - \frac{a_2\sqrt{2g}}{A_2}\sqrt{h_2(t)}\\ y(t) &= h_2(t) + e(t),\end{aligned}$$

where h_i is the water level; A_i and a_i are respectively the cross-sectional areas of the tank and of the outflow orifice; i is the index of the tank, with $i = 1$ for the upper tank and $i = 2$ for the lower tank. g is the standard gravity taken equal to $9.81 m.s^{-2}$. This notation has the advantage of giving physical meaning to the parameters. By looking closely at Fig. 2, the two tanks are really similar. Thus, it is assumed $A = A_1 = A_2$, $a = a_1 = a_2$ and $h^{max} = h_1^{max} = h_2^{max}$. Those relations noticeably reduce the number of parameters. Furthermore, the data provided with the benchmark are recorded in volts (V). That is not a problem for the input because the actuator gain k_u has then $\frac{m^3}{s.V}$ for physical dimension. For the output, we introduce a sensor gain k_s for the output such as $v_2 = k_s h_2$. The voltage v_i is then the image of the water level in tank i. Our new model can be written:

$$\begin{aligned}\dot{v}_1(t) &= -\frac{a\sqrt{2gk_s}}{A}\sqrt{v_1(t)} + \frac{k_u k_s}{A}u(t)\\ \dot{v}_2(t) &= \frac{a\sqrt{2gk_s}}{A}\sqrt{v_1(t)} - \frac{a\sqrt{2gk_s}}{A}\sqrt{v_2(t)}\\ y(t) &= v_2(t) + e(t).\end{aligned}$$

However, this model does not include the overflow from the upper to the lower tank, either the one from the lower tank to the reservoir. To model the overflow, we propose the following model:

$$\begin{aligned}\dot{x}_1(t) &= -\frac{a\sqrt{2gk_s}}{A}\sqrt{v_1(t)} + \frac{k_u k_s}{A}(u(t) - b_u) - k_{over}ov(t)\\ \dot{v}_2(t) &= \frac{a\sqrt{2gk_s}}{A}\sqrt{v_1(t)} - \frac{a\sqrt{2gk_s}}{A}\sqrt{v_2(t)} + k_{over}ov(t)\\ y(t) &= v_2(t) + e(t),\end{aligned}$$

with,

$$\begin{aligned}v_2(t) &\leq v_2^{max}\\ v_1(t) &= \begin{cases} x_1(t), & \text{if } x_1(t) < v_1^{max}\\ v_1^{max}, & \text{otherwise}\end{cases}\\ ov(t) &= \begin{cases} 0 & , \text{ if } x_1(t) < v_1^{max}\\ x_1(t) - v_1^{max}, & \text{otherwise}\end{cases}.\end{aligned}$$

With this model, the integration of $\dot{v}_2(t)$ is saturated at v_2^{max} and $v_1^{max} = v_2^{max} = v^{max}$. In other words, the overflow from the lower tank to the reservoir is not modelled. A bias b_u is added to the input because a better figure of merit (see Appendix A) was observed. In addition, by looking at the measured data, it is assumed $v^{max} = 10V$, considering that the output has neither bias nor scale factor. The set of parameters is $\theta^1_{tanks} = [a \quad A \quad k_u \quad k_s \quad k_{over} \quad v_{1_0} \quad v_{2_0} \quad b_u]^T$ with v_{1_0} and v_{2_0} the voltages of the initial levels in the tanks.

After few trials, it was found that the model contains too many parameters even tough they all have a physical meaning. According to the cross correlation factors (see Appendix B), some of them are indeed linearly linked. The parameters are regrouped according to the following relations: $p_1 = \frac{a\sqrt{k_s}}{A}$ and $p_2 = \frac{k_u\sqrt{k_s}}{a}$. Finally, the model considered for the identification is

$$\begin{aligned}\dot{x}_1(t) &= -p_1\sqrt{2g}\sqrt{v_1(t)} + p_1 p_2(u(t) - b_u) - k_{over}ov(t)\\ \dot{v}_2(t) &= p_1\sqrt{v_1(t)} - p_1\sqrt{v_2(t)} + k_{over}ov(t)\\ y(t) &= v_2(t) + e(t).\end{aligned}$$

The OEM is appropriate because this model is non-linear with respect to the parameters and the states. Furthermore, with the square root function for instance, the derivatives are not defined everywhere. That explains the choice of derivative-free optimisation solvers. Finally, the set of parameters to identify is $\theta^2_{tanks} = [p_1 \quad p_2 \quad v_{1_0} \quad v_{2_0} \quad b_u \quad k_{over}]^T$.

3.2 Identification Results

The cascaded tanks are modelled with Simulink©. The dynamic equations are solved thanks to *ode45* integration solver. Table 1 summarizes the estimated parameters, their relative standard deviations, the figure of merit (see Appendix A) and the computing time for each optimisation solver. The estimation data set contains 1024 points with a sampling time $T_s = 4s$, which is relatively short.

Concerning the initial values of the optimisation solver, v_{2_0} is taken equal to the first recorded output. $v_{1_0} = 5V$ is equivalent to a tank being half full and close to v_{2_0}. The initial bias is neglected. It is assumed that the radius of the tank has an order of magnitude of $1cm$ whereas the one of the outflow orifice is closer to $1mm$. That gives the initial areas. By a real close look at Figure 2, the maximal water level is $h^{max} = 20cm$. Therefore, the initial sensor gain is $k_s^0 = 10/0.2 = 50V/m$. The tank volume can then be estimated close to $60cm^3$. A pump providing a flow of few cm^3/s seems appropriate. Consequently, we take $k_u^0 = 1\frac{cm^3}{Vs}$. Finally, the initial overflow gain arbitrarily set to $k_{over}^0 = p_1^0 p_2^0$ due to the physical dimension. The overflow model is indeed a transfer between a voltage to a voltage per second, like the pump. This initialisation may seem coarse, but a LS initialisation does not suit because the states are not linear with respect to the parameters and h_1 (or v_1) is not available.

Both optimisation solvers almost found equivalent estimated parameters with comparable computing times and figures of merit. The two noteworthy differences are the overflow gain and the relative standard deviations. The large difference for the estimated k_{over} suggests that our overflow model is perfectible. Few trials were undertaken

Table 1. Two tanks identification results

Parameter	Init. Val.	Nelder-Mead	NOMAD
p_1 ($\frac{\sqrt{V}}{\sqrt{m}}$)	0.07	0.0094 (0.52%)	0.0094 (0.0012%)
p_2 ($\frac{\sqrt{m}}{s\sqrt{V}}$)	2.25	4.87 (0.55%)	4.85 (0.0015%)
v_{1_0} (V)	5.00	4.89 (3.1%)	4.96 (0.0021%)
v_{2_0} (V)	5.21	5.16 (2.0%)	5.16 (0.0010%)
b_u (V)	0.0	0.675 (1.8%)	0.663 (0.0012%)
k_{over} $(1/s)$	0.16	0.909 (1.7%)	14.1 (0.0010%)
	e_{RMS}	37.9%	37.6%
	Comp. Time	2min 51s	3min 10s

to improve it without success. The low relative standard deviations found by NOMAD are explained by the fact that this algorithm converged to local optimum where the output sensitivity to the parameters is high. From an algebra point of view, the conditioning number of the jacobian matrix is lower for the NOMAD algorithm than for the Nelder-Mead one. The factor is 10^3. The bad conditioning explains the cross-correlations between some parameters: $\rho^{NM}_{\widehat{p_2}\widehat{b_u}} = 0.99$, $\rho^{NM}_{\widehat{p_2}\widehat{k_{over}}} = 0.96$ and $\rho^{NM}_{\widehat{b_u}\widehat{k_{over}}} = 0.98$. Finally, the parameters estimated with the NOMAD algorithm are more accurate and reliable.

This example shows that, even if the OEM is able to deal with models non-linear with respect to the parameters, the practitioner must be careful with the results. Furthermore, it illustrates the usefulness of keeping the physical meaning of the model, at least at the beginning. That makes the initialisation easier and helps to interpret the results.

4. BOUC-WEN HYSTERESIS

4.1 Model Description

The Bouc-Wen system is a one degree-of-freedom oscillator used in mechanical engineering to represent hysteretic effects. For a complete definition of the hysteresis, please refer to (Schoukens and Noël, 2016) and the references given therein. From the Newton's second law, the Bouc-Wen dynamics is modelled by:

$$m_L\ddot{y}(t) + r(t) + z(t) = u(t), \tag{1}$$

where m_L is the mass, y the output position, u the input force, r the linear restoring force and z the nonlinear force which models the hysteretic memory of the system. The restoring force is modelled by Eq. (2) with the stiffness parameter, k_L, and the viscous damping coefficient, c_L. The hysteretic force, z, is modelled by the dynamic relation (3), where α, β, γ, δ and ν are the parameters defining the shape of the hysteresis. It is worth noting that ν must greater than or equal to 1 in order the system to be stable. From a practical point of view, if the optimisation solver tests the model with non feasible parameters, the simulation solver will diverge. In this case, the optimisation criterion is set to infinity.

$$r(t) = k_L y(t) + c_L \dot{y}(t) \tag{2}$$

$$\dot{z}(t) = \alpha\dot{y}(t) - \beta\left(\gamma\left|\dot{y}(t)\right|\left|z(t)\right|^{\nu-1} z(t) + \delta\dot{y}(t)\left|z(t)\right|^{\nu}\right) \tag{3}$$

According to (Schoukens and Noël, 2016) the Bouc-Wen dynamics is well integrated in time by the Newmark method. This method, developed to a great extent in (Gérardin and Rixen, 2015), is a single-step time integration relevant for second-order differential equations in the field of structural dynamics. It must be noticed that the input force u must be upsampled by a factor 20 to have an accurate integration of the dynamics. Thus, following the integration, the simulated output position y_s must be decimated to the nominal frequency.

Table 2. Bouc-Wen benchmarks parameters

m_L (kg)	c_L $(N.s/m)$	k_L (N/m)	α $(N.s/m)$
2	10	$5\ 10^4$	$5\ 10^4$
β $(N^{\nu-1}/m)$	γ (-)	δ (-)	ν (-)
$1\ 10^3$	0.8	-1.1	1

For this benchmark, the parameters are exactly known; see Table 2. According to (Schoukens and Noël, 2016), the parameters are in S.I. units. For the parameters m_L, c_L, k_L and α, there is no doubt about the units. For the remaining parameters, the authors made the choice to gather the physical dimension on β, in order to keep the other parameter dimensionless. The data are generated with a simulator. This simulator adds, to the output, a band-limited Gaussian (between 0 and 375 Hz) with a root-mean-squared amplitude of $8\ 10^{-3}$ mm. The noise free input signal is multisine containing one steady state period of 8912 samples with an RMS value of 50 N and a sampling frequency $f_s = 750$ Hz. The multisine frequencies are located in the range 50-150 Hz.

4.2 First Results

To perform the OEM, a simulator allowing to change the physical parameters has been developed. This implementation of the Newmark integration method is based on the information provided in (Schoukens and Noël, 2016). The initial parameters are arbitrary chosen to be enough far from the true values while keeping a reasonable computing time; except β which is initialized at its true value for a reason which will be explained below.

The estimated parameters are summarized in Table 3 as well as the relative standard deviations and the relative errors. The results for the NOMAD identification are not presented since there is an issue with this model. All the estimated values are indeed close to the real ones except those for β, γ and δ. Thanks to the knowledge of the true values, it is easy to detect the problematic parameters. If the true values were not available, the larger relative standard deviation could alert the user. In addition, the conditioning number of the jacobian is equal to $1.55\ 10^6$, which is quite large for 8 parameters. Figure 3 illustrates the signals of the jacobian for those three parameters. The sensitivity with respect to β is not observable because of its low order of magnitude. However, it is visible that the sensitivities with respect to δ and γ are similar. From this observation, the linear relation between δ and γ is clear. Finally, the correlation coefficients (see Appendix B) are $\rho_{\widehat{\beta\gamma}} = 0.998$, $\rho_{\widehat{\beta\delta}} = 0.998$ and $\rho_{\widehat{\gamma\delta}} = 1.00$. Such coefficients (greater than 0.95) indicate probable linear relations between those three parameters.

The collinearity between β, γ and δ is now established in practice. We will look for the responsible element of the model. Equation (3) can be written at any sampling time

Table 3. Bouc-Wen identification - First results

Parameter	Init. Val.	Nelder-Mead	Rel. Err.
m_L (kg)	1	1.98 (0.02%)	0.99%
c_L $(N.s/m)$	2	10.3 (0.63%)	3.16%
k_L (N/m)	$1\ 10^4$	$4.95\ 10^4$ (0.10%)	1.03%
α (N/m)	$1\ 10^4$	$4.95\ 10^4$ (0.10%)	0.97%
β $(N^{\nu-1}/m)$	$1\ 10^3$	$3.76\ 10^2$ (8.96%)	62.40%
γ (-)	1.0	2.12 (8.98%)	165.17%
δ (-)	-0.9	-2.93 (8.98%)	165.70%
ν (-)	1.2	1.00 (0.14%)	$2.21\ 10^{-5}$%
e_{RMS} multisine		$4.68\ 10^{-3}$%	
e_{RMS} sinesweep		$2.53\ 10^{-4}$%	
Computing Time		32min 20s	

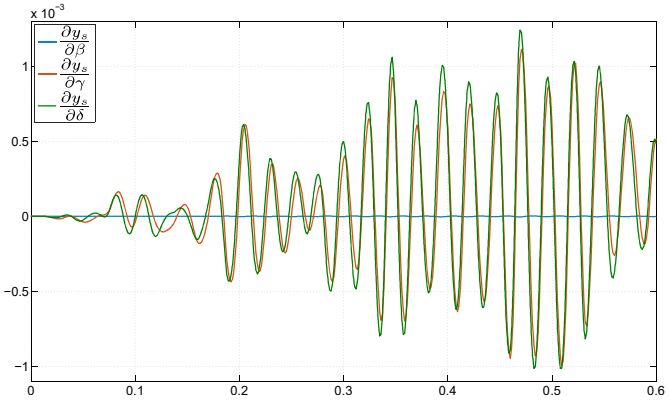

Fig. 3. Time history (zoom) of the sensitivities with respect to: β (light blue), γ (orange) and δ (dashed green)

$$\dot{z} = \alpha f_1(\dot{y}) - \beta\left(\gamma f_2(\dot{y}, z) + \delta f_3(\dot{y}, z)\right). \tag{4}$$

By assuming that the sensitivities with respect to the parameters exist, from that equation, it can be written

$$\begin{aligned}
\frac{\partial \dot{z}}{\partial \beta} &= \alpha\frac{\partial f_1}{\partial \beta} - \gamma f_2 - \delta f_3 - \beta\left(\gamma\frac{\partial f_2}{\partial \beta} + \delta\frac{\partial f_3}{\partial \beta}\right),\\
\frac{\partial \dot{z}}{\partial \gamma} &= \alpha\frac{\partial f_1}{\partial \gamma} - \beta f_2 - \beta\left(\gamma\frac{\partial f_2}{\partial \gamma} + \delta\frac{\partial f_3}{\partial \gamma}\right),\\
\frac{\partial \dot{z}}{\partial \delta} &= \alpha\frac{\partial f_1}{\partial \delta} - \beta f_3 - \beta\left(\gamma\frac{\partial f_2}{\partial \delta} + \delta\frac{\partial f_3}{\partial \delta}\right).
\end{aligned}$$

If the functions f_1, f_2 and f_3 are not sensitive enough with respect to the parameters (i.e. derivatives negligible), it comes out

$$\beta\frac{\partial \dot{z}}{\partial \beta} = \gamma\frac{\partial \dot{z}}{\partial \gamma} + \delta\frac{\partial \dot{z}}{\partial \delta}. \tag{5}$$

Equation (5) makes clear the linear relation between the sensitivities. The assumption that the derivatives of f_1, f_2 and f_3 are negligible is equivalent to say that $\dot{z}$ is linear with respect to α, $\beta\gamma$ and $\beta\delta$. That kind of assumption is called Pseudo Linear Regression (PLR) in system identification, see Eq. (7.112) in (Ljung, 1999). To identify the model, coefficients $\gamma' = \beta\gamma$ and $\delta' = \beta\delta$ could be introduced. However, to make easier the interpretation of the parameters, β will be kept constant to its true value without being estimated. This is why we did not try to initialised β to another value at the beginning.

4.3 Final Results

A second identification process is performed with β fixed to its true value to avoid any modelling error. The results are summarized in Table 4. Both optimisation solvers provided close estimates with similar relative standard deviations. The figures of merit are comparable and small. The identified models can be considered as satisfactory. The main difference lays in the computing time. Surprisingly, the Nelder-Mead simplex significantly took less time than the NOMAD algorithm. Therefore, for this example, the Nelder-Mead optimizer appears to be more appropriate.

This second example confirmed that the OEM is able to deal with models non-linear with respect to the parameters. However, it also confirmed that the practitioner must be careful between the model and the potential links between the parameters. Finally, it proved that the optimisation can be performed with derivative-free solvers, but the jacobian matrix is unavoidable to analyse the results.

5. CONCLUSION

In this article, two non-linear systems were identified thanks to the output error method. This method relies on the simulation of continuous-time systems and derivative-free optimisation solvers. From a general point of view, the method gives satisfactory results. In addition, we have learnt that:

- even though the method is able to deal with non-linear models, the user must be careful with the parametrization;
- it is useful to keep a physical meaning in order to analyse the model;
- even though the identification problem can be solved thanks to derivative-free optimizer, the sensitivities are the key element of the results analysis.

REFERENCES

Abramson, M., Audet, C., Couture, G., Dennis, Jr., J., Le Digabel, S., and Tribes, C. (2016). The NOMAD project. Software available at `https://www.gerad.ca/nomad/`.

Carrillo, F., Baysse, A., and Habbadi, A. (2009). Output error identification algorithms for continuous-time systems operating in closed-loop. *IFAC Proceedings Volumes*, 42(10), 408–413.

Currie, J. and Wilson, D.I. (2012). OPTI: Lowering the Barrier Between Open Source Optimizers and the Industrial MATLAB User. In N. Sahinidis and J. Pinto (eds.), *Foundations of Computer-Aided Process Operations*. Savannah, Georgia, USA.

Garnier, H. and Wang, L. (2008). *Identification of continuous-time models from sampled data*. Springer.

Garnier, H. and Young, P.C. (2014). Special issue on applications of continuous-time model identification and estimation. *International Journal of Control*.

Gautier, M., Janot, A., and Vandanjon, P.O. (2013). A new closed-loop output error method for parameter identification of robot dynamics. *IEEE Transactions on Control Systems Technology*, 21(2), 428–444.

Gérardin, M. and Rixen, D.J. (2015). *Mechanical vibrations: theory and application to structural dynamics.*

Table 4. Bouc-Wen identification - Final results

Parameter	Init. Val.	Nelder-Mead	Rel. Err.	NOMAD	Rel. Err.
m_L (kg)	1	1.98 (0.02%)	0.78%	1.98 (0.02%)	0.81%
c_L $(N.s/m)$	2	8.98 (0.75%)	10%	9.12 (0.73%)	8.8%
k_L (N/m)	$1\ 10^4$	$4.94\ 10^4$ (0.10%)	1.2%	$4.95\ 10^4$ (0.10%)	1.0%
α (N/m)	$1\ 10^4$	$4.97\ 10^4$ (0.10%)	0.62%	$4.96\ 10^4$ (0.10%)	0.84%
γ (-)	1.0	$8.13\ 10^{-1}$ (0.60%)	1.6%	$8.13\ 10^{-1}$ (0.60%)	1.6%
δ (-)	-0.9	-1.12 (0.63%)	1.1%	-1.13 (0.62%)	1.2%
ν (-)	1.2	1.00 (0.15%)	$4.8\ 10^{-5}$%	1.00 (0.15%)	$7.8\ 10^{-7}$%
e_{RMS} multisine		$4.68\ 10^{-3}$%		$4.68\ 10^{-3}$%	
e_{RMS} sinesweep		$1.86\ 10^{-4}$%		$1.90\ 10^{-4}$%	
Computing Time		44min		4h 23min	

John Wiley & Sons, Chichester, United Kingdom, 3rd edition.

Hairer, E., Nørsett, S.P., and Wanner, G. (1993). *Solving Ordinary Differential Equations I*, volume 8. Springer-Verlag, Berlin, 2 edition.

Jategaonkar, R. (2006). *Flight vehicle system identification: a time domain methodology*, volume 216. AIAA, Reston, VA, USA.

Johnson, S.G. (2016). The NLopt nonlinear-optimization package. Software available at `http://ab-initio.mit.edu/nlopt`.

Klein, V. and Morelli, E.A. (2006). *Aircraft system identification: theory and practice*. American Institute of Aeronautics and Astronautics Reston, Va, USA.

Ljung, L. (1999). *System identification: Theory for the User*. Prentice Hall, 2 edition.

Nelder, J.A. and Mead, R. (1965). A simplex method for function minimization. *The computer journal*, 7(4), 308–313.

Schoukens, M. and Noël, J.P. (2016). IFAC world congress 2017 special track proposal on nonlinear system identification benchmarks. .

Walter, E. and Pronzato, L. (1997). *Identification of parametric models from experimental data*. Springer Verlag.

Wigren, T. (2006). Recursive prediction error identification and scaling of non-linear state space models using a restricted black box parameterization. *Automatica*, 42(1), 159–168.

Appendix A. FIGURE OF MERIT

(Schoukens and Noël, 2016) provide a set of estimation data to identify the model and a set of test (or validation) data to assess the identified model. To compare the different methods, the following figure of merit is given:

$$e_{RMS} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(y_s(t_k) - y_{val}(t_k)\right)^2} \tag{A.1}$$

with N the number of sampling points, y_val the validation output and y_s the simulated output from the identified model.

Appendix B. STATISTICAL ACCURACY

It is assumed that there are no modelling errors and that the estimator is efficient. Considering a system with a single output, the covariance matrix of the estimated parameters is then given by

$$P = \sigma^2 \left[\frac{\partial \boldsymbol{y}_s}{\partial \widehat{\boldsymbol{\theta}}}^T \frac{\partial \boldsymbol{y}_s}{\partial \widehat{\boldsymbol{\theta}}} \right]^{-1}, \tag{B.1}$$

where σ^2 is the covariance of the output noise and $\frac{\partial \boldsymbol{y}_s}{\partial \widehat{\boldsymbol{\theta}}}$ is the output sensitivity matrix at the convergence point. In fact, P is the Cramer-Rao lower bound, which is reached if the estimator is efficient. For further details see e.g. (Walter and Pronzato, 1997). The standard deviation of the i^{th} estimated parameter is

$$\sigma_{\widehat{\theta}_i} = \sqrt{P_{ii}}. \tag{B.2}$$

The relative standard deviation is then $100 * \sigma_{\widehat{\theta}_i} / \left|\widehat{\theta}_i\right|$ for non zero estimated parameters. Finally, the correlation coefficients, which express the statistical dependence between the estimated parameters, are calculated as follows:

$$\rho_{\widehat{\theta}_i \widehat{\theta}_j} = \frac{P_{ij}}{\sqrt{P_{ii}}\sqrt{P_{jj}}} \tag{B.3}$$

According to (Jategaonkar, 2006), a correlation coefficient greater than 0.95 requires some attention from the practitioner since it may be the result of a linear dependence between the two considered parameters.